Amendments to Claims

The listing of claims will replace all prior versions, and listings of claims in the application.

1. (Currently Amended) An authentication engine architecture for a SHA-1 multi-round authentication algorithm, comprising:

a hash engine configured to implement hash round logic for a SHA-1 authentication algorithm, said hash round logic implementation including,

a combined adder tree with a timing critical path having a single 32-bit carry look-ahead adder (CLA), wherein said combined adder tree has two parallel outputs.

2. (Currently Amended) The authentication engine architecture of claim 1, wherein said hash round logic implementation has a timing critical path equivalent to one of:

one 5-bit addition, one 32-bit carry save adder (CSA), a multiplexer operation, and one 32-bit CLA; and

three 32-bit CSAs, a multiplexer operation, and one 32-bit CLA.

3. (Original) The authentication engine architecture of claim 1, wherein the additions performed by the combined adder tree are preceded by a 5-bit circular shifter.

4. (Original) The authentication engine architecture of claim 3, wherein combined adder tree includes add5tol and add4tol adders.

5. (Original) The authentication engine architecture of claim 1, wherein the combined adder tree is configured such that addition computations are conducted in parallel with round operations.

6. (Original) The authentication engine architecture of claim 1, wherein the architecture is implemented as an authentication engine architecture for a multi-loop, SHA-1 authentication algorithm, comprising:

a first instantiation of a SHA-1 authentication algorithm hash round logic in an inner hash engine;

a second instantiation of a SHA-1 authentication algorithm hash round logic in an outer hash engine;

a dual-frame payload data input buffer configured for loading one new data block while another data block one is being processed in the inner hash engine;

an initial hash state input buffer configuration for loading initial hash states to the inner and outer hash engines for concurrent inner hash and outer hash operations; and

a dual-ported ROM configured for concurrent constant lookups for both inner and outer hash engines.

7. (Currently Amended) The authentication engine architecture of claim 6, wherein the multi-loop, multi-round authentication algorithm is HMAC-SHA-1.

8. (Currently Amended) The authentication engine architecture of claim 1, wherein said hash round logic is implemented such that eighty rounds of a SHA-1 loop are collapsed into forty rounds.

9. (Currently Amended) The authentication engine architecture of claim 1, wherein said hash engine is configured to implement hash round logic comprising:

five hash state registers;

one critical and four non-critical data paths associated with the five registers, such that in successive SHA-1 rounds, registers having the critical path are alternative.

10. (Currently Amended) A method of authenticating data transmitted over a computer network, comprising:

receiving a data packet stream;

splitting the packet data stream into fixed-size data blocks; and

processing the fixed-size data blocks using a SHA-1 multi-round authentication engine architecture, said architecture implementing hash round logic for a SHA-1 authentication algorithm including a combined adder tree with a timing critical path having a single 32-bit carry look-ahead adder (CLA), wherein said combined adder tree has two parallel outputs.

11. (Original) The method of claim 10, wherein the hash round logic implementation has a timing critical path equivalent to one of:

one 5-bit addition, one 32-bit CSA, a multiplexer operation, and one 32-bit CLA; and.

three 32-bit CSAs, a multiplexer operation, and one 32-bit CLA.

12. (Original) The method of claim 10 wherein additions performed by the combined adder tree are preceded by a 5-bit circular shifter.

13. (Currently Amended) The method of claim 10, further comprising:

providing five hash state registers; and

providing data paths from said five state registers such that four of the five data paths from the registers in any SHA-1 round are not timing critical.

14. (Currently Amended) The method of claim 13, wherein, in successive SHA-1 rounds, registers having the critical path are alternative.

15. (Currently Amended) The method of claim 14, wherein eighty rounds of a SHA-1 loop are collapsed into forty rounds.

16. (Original) The method of claim 10, wherein addition computations are conducted in parallel with round operations.

17. (Original) The method of claim 10, wherein said authentication engine is a multi-loop, multi-round authentication engine architecture having a hash engine core

comprising an inner hash engine and an outer hash engine, said architecture configured to,

pipeline hash operations of said inner hash and outer hash engines,

collapse and rearrange multi-round logic to reduce rounds of hash operations, and

implement multi-round logic such that addition computations are conducted in parallel with round operations.

18. (Currently Amended) The method of claim 17, wherein the multi-loop, multi-round authentication algorithm is HMAC-SHA-1.

19. (Original) The method of claim 18, wherein said pipelining comprises performance of an outer hash operation for one data payload in parallel with an inner hash operation of a second data payload in a packet stream fed to the authentication engine.

20. (Original) The method of claim 19, wherein a dual-frame input buffer is used for the inner hash engine.

21. (Original) The method of claim 20, wherein initial hash states for the hash operations are double buffered for concurrent inner hash and outer hash operations.

22. (Currently Amended) The method of claim 21, wherein concurrent constant lookups are performed from a dual-ported read only memory (ROM) by both inner and outer hash engines.

23. (Previously Presented) The authentication engine architecture of claim 1, comprising a multiplexer to select an output of the combined adder tree.

24. (Previously Presented) The method of claim 10, comprising performing a multiplexing operation to provide an output of the combined adder tree.